Available online at www.sciencedirect.com
ScienceDirect
Procedia Engineering 100 (2015) 1459 – 1468
Procedia Engineering
www.elsevier.com/locate/procedia

25th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAAM 2014

Formation of Control Structures in Static Swarms

Valery Karpov, Irina Karpova*

National Research University Higher School of Economics, 20 Myasnitskaya Ulitsa, Moscow, 101000, Russia

Abstract

Work solutions are proposed for problems of leader definition and role distribution in homogeneous groups of robots. It is shown that transition from a swarm to a collective of robots with hierarchical organization is possible using exclusively local interaction. The local revoting algorithm is central to the procedure for choice of leader while redistribution of roles can be achieved by a wave method. The basis for this approach is the static swarm model, which is characterized by the absence of a set control center and represents the network fixed at some time interval as a set of locally interacting agents.

Peer-review under responsibility of DAAAM International Vienna

Keywords: *swarm robotics; voting procedur; selecting a group leader; roles distribution; local interaction*

1. Introduction

Active research into the creation of systems of interacting robots has been ongoing for nearly a quarter of a century. Approaches such as collective, swarm and flocking robotics have been prominent in modern robotics and the theory of multi-agent systems, but the overwhelming tendency of research in this area remains at a theoretical, model level. According to numerous reviews [15, 17], it is apparent that this absence of valid, significant results is not least connected with the relative neglect of a number of important tasks. Research in the field of group robotics (to coin a generalized name for collective, swarm, flock etc. robotics) has a very fragmentary character.

One warning sign is the obvious transfer of the center of gravity of research to swarm robotics, which can be viewed as a simpler, more basic level of group robotics model. Among a set of problems in swarm robotics that remain insufficiently studied, two will be elaborated here. The first of these is the problem of leader definition in a

* Corresponding author. Tel.: +7-916-143-68-09.
E-mail address: vkarpov@hse.ru, ikarpova@hse.ru

Peer-review under responsibility of DAAAM International Vienna
doi:10.1016/j.proeng.2015.01.517

homogeneous group of robots, and the second is role distribution among members of the group under conditions of exclusively local interaction.

Leadership. One of the basic features of swarm robotics is the local character of interaction of robots, with each other and with their environment [17]. This kind of interaction is called implicit communication [15], which means that each robot in the group interacts directly only with neighbours within some limited visibility range.

In such systems, it usually follows that robots make decisions independently on further actions, guided by some simple rules of local interaction. However, the overwhelming majority of examples of task solution in the field of swarm robotics concern the coordinated movement of a swarm. For instance, in the obvious and rather simple task characterised as the “Leader-Follower” method in [13], it is considered that there is an a priori leader in the group who sets this movement.

There are many variants of local interaction rules, from the formalistic [12] to the very exotic. For instance, [3] describes the virtual spring-damper model of flocking mobile robots’ behavior. The “spring” component of the model defines the force of attraction to the leader (not follower to follower), and the “damper” component defines the repelling force.

Another especially technical approach has been proposed in [4], in which one of the robots must become the 'leader' of the group by maximizing the value of its communicative output. The main issue of this method is that it leads to appearance of a set of leaders, and a number of them depends on the topology of the swarm. A similar technical approach was offered in [6], in which the agent with the greatest weight was appointed as leader. The difference between the leader and other members of the group (or a flock) was that the leader did not use the rule “move to the nearest neighbour”. This approach provided a solution to the problem of coordinated movement, turning a swarm into a flock. In a very interesting project described in [11], autonomous robots used rather simple “Fish Behavior” interaction rules for collision-free driving, in which all robots were equipped with a set of complex sensors.

This paper deals mainly with the a priori set leader, or with techniques that avoid defining the leader under the conditions of some specific objective. However, in a number of works, a swarm leader is elected. In [8], group leader selection is based on optimizing power consumption, making it necessary to know the distances between robots and the power consumed by transfer of the message from one robot to another. In [16], swarm traffic control using the center of masses or the geometrical centre of a swarm is described. This method requires that coordinates of agents, their speed and their direction of movement must be known. Another paper [10] describes a localized mechanism for determining the information potential on each node, based on local process information and the potential of neighbouring nodes. In that instance, the node with the minimum potential was considered to be the leader, but we are interested here in the problem of leader identification in a more general case, in terms of mechanisms of local information interaction.

Role distribution. For solution of the basic problem of the coordinated movement of robots, it is sufficient that there is a leader. However, more complex challenges solved by a group of robots require differentiation of their functions and, generally speaking, distribution of tasks between robots; this is perhaps one of the most problematic concepts of swarm robotics. In reviews mentioned above [15, 17], task distribution in robot groups was considered rather declaratively—at best, some physical models, methods of distributed planning, optimization and other general mechanisms are mentioned [5]. In practice, one usually deals with either the centralized control systems or with homogeneous groups without functional differentiation. For example, [9] describes an assembly system model with self-organizing behaviour (Bionic Assembly System—BAS). The behaviour of each mobile robot in this system depends on its internal state and on the state of the system. A central computer plans the global production of BAS, synchronizing the supply of parts and so on, because differentiation of functions and distribution of tasks is not considered an actual problem by swarm robotics. Instead, it is usually taken that the swarm has to solve only simple, mass problems like coordinated movement. This certainly reduces the importance of the swarm approach and goes straight to the main declared thesis of swarm robotics as an approach to the solution of complex behavioural tasks using a set of simple technical devices—robots.

In the present case, it is considered that the problems of leader formation and role distribution are extremely important for the development of swarm robotics. This paper considers some ways in which these tasks might be solved, using such model the organization of a group of robots, as a static swarm.

2. Task definition

The task is formulated as follows. Consider a set of simple devices (robots or agents) capable of directing local interaction between neighbours. The question is whether it is possible to formulate conditions under which it will be possible for such a system to solve more complex problems, both at the behavioural level and at the levels of information processing, decision-making, and so on. In other words, it is necessary to define the conditions of emergence of synergetic effects or emergent properties.

There are two main classes of swarm models: micro- and macro-level models. The first of these is based on the models of the behaviour of individuals; the second class is based on description of the swarm as a whole. For example, at the micro-level, models of finite-state machines are widely used, and at the macro-level, hydrodynamics models are common. Hybrid models combining both approaches are less often applied. For example, Berman et al. describe a model of the dynamics of environment state in [1] that defines the behaviour of members of a swarm (agents). The reasoning in the present study generally belongs to the micro-level class because our interest is in the mechanism of local interaction of robots in a swarm (as in [14]).

To begin, consider the structure of an agent, where the main objective is a set of some simple devices. Simplicity here means some principal limitation of cognitive abilities (sensors, calculations and memory).

Further, some serious restrictions will be placed on the robots' communication opportunities, in which each robot can communicate with no more than some limited number of its neighbors. It will further be assumed that the robots have a fixed number of communications ports (i.e. contact points that form information channels)—for example, robots must be connected physically to each other's communication ports for communication as organized in Fig. 1.

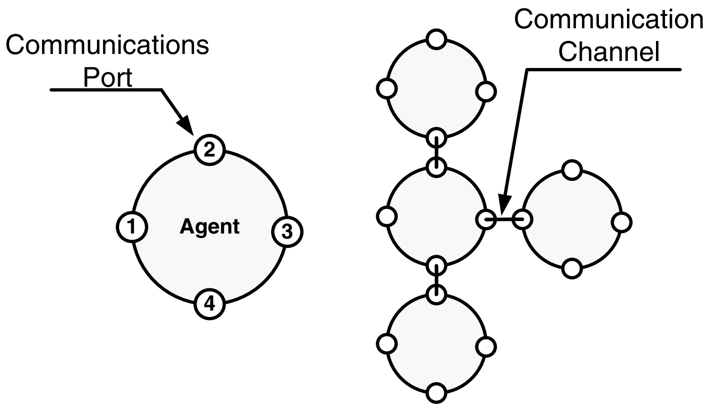

Fig. 1. Robots with four communications ports and their communications.

3. Static Swarm

One of the models describing the organization of a great number of locally interacting agents or robots is the so-called *static swarm*. This arrangement is characterized by the absence of a control centre and represents the given network fixed at some time interval as a set of agents [7].

The main feature of a static swarm is that at some moment, instead of a set of separated agents, the computing structure is completely defined, enabling the solution of difficult calculating and data processing tasks. It follows that a static swarm can be considered as an object with qualitative properties other than a simple set of agents.

The main properties of a static swarm are an activity, locality of interactions and functional heterogeneity.

Activity. Certainly, unlike the computer network, the network of agents has to be capable of perceiving signals from the environment, and of producing some effector functions (such as motion, for example) in order to have an impact on the outside environment.

Locality of interaction. An important feature of a static swarm is essentially the local nature of interaction: agents communicate only with their neighbours.

Functional heterogeneity. The solution of complex tasks (i.e. manifestation of emergent properties of a system) assumes heterogeneity exists in the group in terms of differentiation of functions carried out by agents: strategic and tactical management, gathering and information processing, realization of effector functions, and so on. The organization of the mechanism of this functional heterogeneity is therefore an important question.

The following task will be considered. Let there be a set of agents (robots) capable of local information exchange between nearest neighbours. Further, at some timepoint, the static swarm must realize a certain procedure for role distribution; some individual must act as a control centre, another must serve the data processing function, another has to collect information from the external environment, and so on.

The general principles of role distribution can be based on the following obvious reasoning: a node (agent) with a maximum number of links (neighbours) becomes a candidate for the control centre role. Its nearest neighbours are analyzers of information; they prepare information for decision-making. Nodes located on the periphery of a network are responsible for gathering information. An example of such a network of agents is shown in Fig. 2.

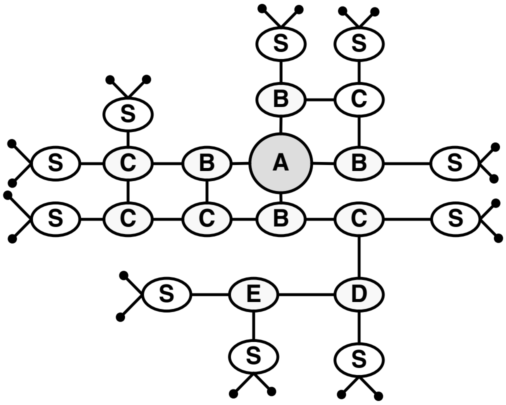

Fig. 2. An example of the organization of a network.

Here, node A becomes a main control center; its nearest neighbours (B) are assigned the role of analyzer; and peripheral nodes (S) will act as external sensors. Labels C, E and D designate other roles. The main question, then, is how these node-agents will choose the central, main node. There are several possible ways to organize such a vote.

4. Voting task

Consider the following formulation of the voting task. Let there be a set of agents with limited communication opportunities, so that agents are capable only of directing local interactions between neighbours. The task requirement is that agents must choose a leader by voting.

All further assumptions are based on the fact that the topology of a network is unknown and that all reasoning must have an especially "local" character (i.e. considered from the point of view of the agent taking part in voting).

At some timepoint, agents receive a global signal that the vote is beginning. At this moment, each agent establishes communication channels with its neighbours, and a generally directed graph is formed. The graph's node is an agent; incoming edges are interpreted as the ability to receive data from source nodes. In this way, communication channels are formed.

Let a static swarm be fixed (i.e. assume that its topology will not change further).

Each agent is described as

$$A = (\alpha, L, C, W)$$

where $\boldsymbol{\alpha}$ = agent's identifier or name; L = a list of agent-neighbors from whom the agent $\boldsymbol{\alpha}$ can receive information (incoming edges); C = a candidate's identifier, for whom the agent $\boldsymbol{\alpha}$ votes; W = weight of candidate C, i.e. the number of votes which, in the agent's opinion, should be given to the candidate.

The essence of a voting procedure is that each agent defines for whom its neighbours vote. Depending, then, on the weight of the candidate for whom the neighbour votes, the agent can change its opinion and vote for the same candidate.

Figure 3 presents one step of this voting scheme. The node labels indicate the following: the agent's identifier $\boldsymbol{\alpha}$ is the "numerator", and values C_a and W_a represent the candidate's identifier and weight.

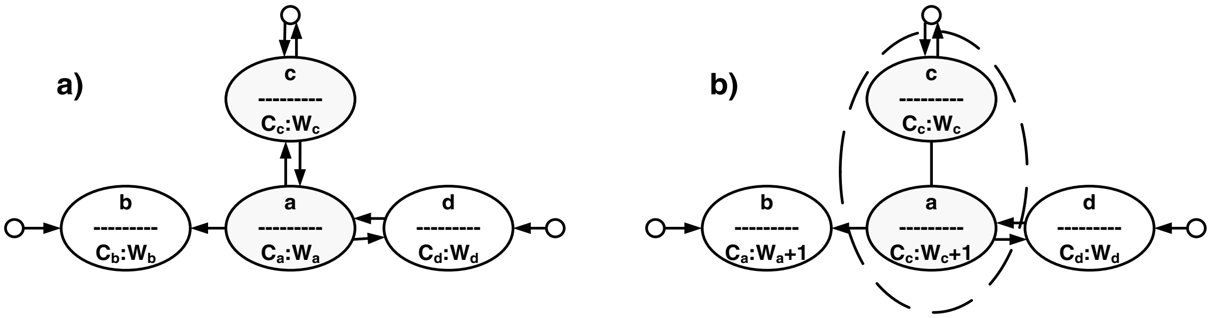

Fig. 3. One voting step: a) = start condition, b) = votes final distribution.

Assume that agent $\boldsymbol{\alpha}$ votes for candidate C_a and agent $\boldsymbol{c}$ votes for candidate C_c. If weight W_a is less W_c then agent $\boldsymbol{\alpha}$ can change its opinion and revote, adding one more voice to the weight of the new candidate.

The probability that agent i will change their opinion under the influence of the opinion of agent j (an opponent) can be defined as follows:

$$p_{ij} = \frac{W_i}{W_i + W_j}$$

That is, the tendency to change opinion naturally depends on the degree of conviction or weight of the candidate.

The distribution of voices of candidates and their weight at an initial timepoint is also implemented quite naturally; each agent votes for itself (declares itself the candidate), and the weight of this decision is equal to the number of this agent's neighbours.

Algorithms for the agent's voting behaviour are given below.

Algorithm G1(α). Agent's decision-making—1

$\boldsymbol{\alpha}$—agent
C_a—the candidate for whom agent $\boldsymbol{\alpha}$ votes
W_a—candidate's weight
L_a—list of agent's candidates

Procedure G1($\boldsymbol{\alpha}$)

To choose among neighbours of the opponent with maximum weight A_{op}:

$$A_{op} \in L_\alpha, C_{op} \neq C_\alpha$$
$$W_{op} = \max_{i \in L_\alpha} W_i$$

To calculate value of probability of change of opinion:

$$p_\alpha = \frac{W_{op}}{W_\alpha + W_{op}}$$

To change opinion with probability p_a:

$$C_a \leftarrow C_{op}$$
$$W_a \leftarrow W_{op} + 1$$

end procedure G1(***α***)

It is possible to “roughen” the decision-making algorithm, forcing the agent to change its opinion on the candidate if there is stronger opponent in its environment. If parity is observed between scales of opinion of the agent and the strongest opponent, the choice of decision can be carried out probabilistically.

Algorithm G2(*α*). Agent’s decision-making—2

Procedure G2(***α***)
To choose among neighbours of the opponent with the maximum weight A_{op}:

$$A_{op} \in L_\alpha, C_{op} \neq C_\alpha$$
$$W_{op} = \max_{i \in L_\alpha} W_i$$

if $W_{op}>W_a$ then -- The opponent is "stronger". We change the opinion:

$$C_a \leftarrow C_{op}$$
$$W_a \leftarrow W_{op} + 1$$

else
if $W_{op}=W_a$ then -- Forces are equal. We change the opinion -- with probability 0.5
p ← rand() -- Random value from 0 to 1
if p > 0.5 then

$$C_a \leftarrow C_{op}$$
$$W_a \leftarrow W_{op} + 1$$

end if
end if
end procedure

A common voting scheme might look like this:

Algorithm V(*A*). Voting
A—a set of agents

```
α—agent
Ca—the candidate for whom the agent α votes
Wa—candidate's weight
La—list of agent's candidates
eoj—flag of end of a voting procedure
procedure V(A)
  eof ← false
  for all α∈A do                -- Agents initialization
```

$$C_a \leftarrow \alpha$$

$$W_a \leftarrow \dim(L_a)$$

```
  end for
  while not eoj do              -- Main cycle of vote
    for all α∈A do              -- Cycle on all agents
      G1(α)
    end for
    "Definition of conditions of completion of voting procedure eoj"
  end while
end procedure V(A)
```

In this algorithm, the biggest problem is the item "Definition of conditions of completion of voting procedure eoj". In the absence of global information on a network's state, the agent has to make the decision for itself that voting is finished. Information received from the immediate environment is obviously not sufficient for this purpose, and two variants of agent behaviour are therefore possible:

1. To consider that voting must be completed in at most some certain number of steps, involving top assessment of the number of voting algorithm steps.

2. To realize some procedure for an exchange of messages defining that voting is completed, and no agent further changes their decision.

The first variant must prove convergence of iterative voting procedures. Some reasoning can be based by analogy with a schema for reaching a consensus [2]. DeGroot's paper defines consensus as mutual agreement on a subject among a group of people (agents in our terminology). The main issue is that with DeGroot's schema convergence can be proved only for some partial situations.

The second variant also implies the existence of some assessed number of voting steps prompting the agent to send a request defining voting procedure completion. The realization of procedures of this sort also presents a number of highly technical difficulties—in particular, an increase in network traffic, as each agent must realize this procedure irrespective of the others.

As a further example of a voting procedure, Figure 4 shows 3 steps in the voting procedure for a solid group of robots.

In the first step, each agent votes for itself, so that the number of cells designating "borders" of distribution of voices for the corresponding candidates is equal to the number of robots. The second step (revoting) shows an integration of areas voting for chosen candidates. Finally, in the sixth step, all votes are assigned to a single candidate, and the voting procedure comes to the end. An example of the process of voting in extremely adverse conditions is shown in Figure 5, involving two clearly expressed zones connected by two isthmuses.

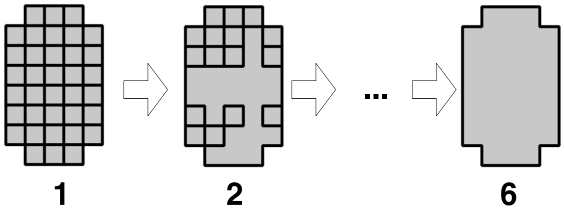

Fig. 4. A voting procedure in solid group. Steps 1, 2 and 6.

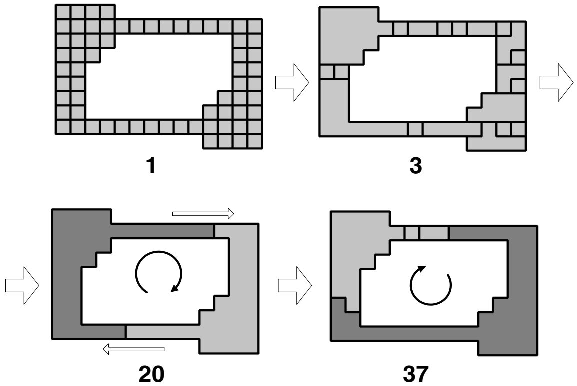

Fig. 5. Cycling voting procedure. Steps 1, 3, 20 and 37.

In this situation, a cyclic process of distribution of voices can be observed. At the 20th step of the vote, two stable areas are formed, each of which votes for their own candidate, and a further process of cyclic revoting begins. It is clearly visible how preference areas actually trade places in the 37th step. The process continues only as far as step 51, when all oscillations stop and a single candidate remains.

Justification of these algorithms requires answers to two main questions: (1) convergence of the algorithms to one solution and (2) estimation of the number of voting steps. Unfortunately, these questions currently remain open, as we can speak only about the results of modelling, according to which the process of voting converges. Clearly, the number of voting steps did not exceed the number of robots in the group.

Centralized voting. Problems with the schemes above result in particular from the local nature of the agents' decision-making. If each agent knew the graph structure, definition of the leader would be quite a routine task. In fact, it is possible to provide a rather simple scheme of an exchange of messages between agents, which with guarantee would allow receipt of the full graph structure. The number of steps does not exceed the number of robots N in a group. To achieve this, it is enough for the robots to report to each other everything they know about the structure of the graph at the present time, as follows:

At an initial timepoint, information about the structure of the graph for each agent is limited to knowledge of its neighbours. This incomplete graph is represented, for example, by the list of edges L^i_0 sent by agent i to its neighbours. Having received such a list, each agent combines it with their existing list for a fuller picture in the form of a new list:

$$L_t^i = \bigcup_{k \in Z} L_{t-1}^k$$

This is a combination of the lists received from all neighbours from some area Z at the previous timepoint.

Through no more than N steps, then, each robot will have all the information about the graph's structure. Further, all of them elect the only leader, proceeding, for example, from reasons of maximum connectivity, equidistance and so on. An obvious and ineradicable deficiency of such a scheme is a very big flow of information, which should be reported to robot neighbours. The practicality of this scheme in real systems is rather doubtful.

5. Task distribution

In the absence of morphological distinctions between agents, role distribution in a static swarm is defined exclusively by the current topology of the system. The distribution process is presented as a well-known procedure of control wave distribution. The initiator of distribution is the leader, whose role is designated as R_0. Direct neighbours of the leader receive an initial message, according to which a role R_1 is assigned to them, and so on. Thus, the role of robot i is defined by the roles in its environment:

$$R_i = \max_{k \in Z} R_k + 1$$

The roles wave distribution is realized exclusively by local interaction, but there is one essential problem. For successful functioning of system let M roles be required, with the process of distribution of a wave consisting of L steps (Figure 6).

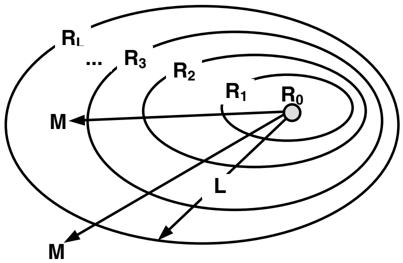

Fig. 6. Actual (L) and required role number (M).

If $M = L$, there is no problem. If $M < L$, there are too many performers with role R_M. This is not a good situation, but it is not fatal because in a static swarm we are not interested in role distribution optimization (like [5]). If $M > L$, however, the situation is worse, as there is a deficiency of performers, which is extremely undesirable. Agents playing several roles at once (combination of specializations) can cover the performance deficit, so that definition of the procedure whereby an agent should assume additional functions seems simple. For example, if an agent with a role number L has no neighbours with roles bigger than L, it means that this agent is at the periphery. Further, if we know that M roles are needed, this agent has to assume roles from L to M. The prevention of any deficiency can also be defined in advance. If there is a group of N agents with maximum connectivity to each agent s (the maximum number of neighbours), it is possible to estimate the minimum number of roles M. Estimation of the M value is

$M \sim \log_S N$

6. Conclusions

Simple and effective methods have been proposed for the solution of important problems of swarm robotics such as leader definition and role distribution in a group of agents. Efficiency is understood as the acceptability of robots with limited cognitive abilities (insufficiency of sensory abilities, computing capacities, communications channels etc.—in short, all that is peculiar to a swarm robotic). Despite their simplicity, realization of these mechanisms confirms the basic possibility of formation of very complex structures in the organization of homogeneous groups, and again confirms that distinctions between a swarm, a flock and a collective of robots are somewhat artificial.

The static swarm model is a convenient way of looking at swarm robot organization. While it is limited to exclusively local interaction between agents, it offers all the advantages of a system understood as a network of connected agents, allowing solution of such problems as storage and data processing, coordinated movement and so on [7].

In future work, we hope to investigate the mechanism of logical consequence in static swarms, hypothesizing that logical consequence procedures can be implemented by exclusively local interaction methods.

Acknowledgements

We gratefully acknowledge the support of RFBR grant No.14-01-00817.

References

[1] Berman S., Halasz A., Kumar V., Pratt S.: Bio-Inspired Group Behaviors for the Deployment of a Swarm of Robots to Multiple Destinations. In 2007 International Conference on Robotics and Automation. pp.2318-2323, IEEE, 2007.

[2] DeGroot M. H., Reaching a Consensus. J. Amer. Stat. Assoc., 1974, Vol 69, Issue 345, pp.118-121.

[3] Dewi T., Risma P., Oktarina Y. Wedge Formation Control of Swarm Robots. The 14th Industrial Electronics Seminar, Electronic Engineering Polytechnic Institute of Surabaya (EEPIS), Indonesia, 2012, pp.294-298.

[4] Gigliotta, O., Mirolli, M., Nolfi, S. Communication based dynamic role allocation in a group of homogeneous robots. Natural Computing, Volume 13, Issue 3, September 2014, pp. 391-402.

[5] Kalyaev I., Kapustjan S., Ivanov D. Decentralized control strategy within a large group of objects based on swarm intelligence. Proc. of the 2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics, RAM 2011, pp.299-303.

[6] Karpov, V.E. Partial mechanisms of leadership and self-consciousness in swarm robotics. Proc. of the 13th National Conf. on Artificial Intelligence KII-2012, Belgord, 2012, (Belgorod State Technical University, Belgorod, 2012), Vol. 3, pp.275-283 [in Russian].

[7] Karpov, V.E. Control in static swarms. Problem definition. Proc. of the VII International science-practical conference on Integrated models and soft computing in artificial intelligence, Kolomna. (Fizmatlit, Moscow, 2013). Vol. 2, pp.730-739 [in Russian].

[8] Kim, S.-C., Shin, K.H., Woo, C.-W., Eom, Y.S., Lee, J.M. Performance analysis of entropy-based multi-robot cooperative systems in a MANET. International Journal of Control, Automation and Systems, Volume 6, Issue 5, October 2008, pp.722-730.

[9] Kukushkin, I.K.; Katalinic, B.; Cesarec, P.; Zdyb, D. & Kettler, R.: Modeling of mobile robot behavior for line-less Bionic Assembly System, Annals of DAAAM for 2012 & Proceedings of the 23rd International DAAAM Symposium, ISBN 978-3-901509-91-9, ISSN 2304-1382, pp. 0865-0870, Editor B[ranko] Katalinic, Published by DAAAM International, Vienna, Austria, 2012.

[10] Loukas Andreas, Woehrle Matthias, Glatz Philipp, Langendoen Koen. On Distributed Computation of Information Potentials. Proceedings of the 8th ACM SIGACT/SIGMOBILE International Workshop on Foundations of Mobile Computing, FOMC'12. 2012, Madeira; Portugal; 19 July 2012, Article number 5.

[11] Nissan EPORO Robot Car "Goes to School" on Collision-free Driving by Mimicking Fish Behavior - Advanced Robotic Concept Debuts at CEATEC JAPAN 2009. – http://www.nissan-global.com/EN/NEWS/2009/_STORY/091001-01-e.html.

[12] Pavlovsky V.E., Kirikova E.P., Pavlovsky V.V. Modeling of behavior of big groups of robots in the environment with obstacles. Proc. of the scientific seminar "Control in distributed network-centric and multi-agents systems". St.Petersburg: "Koncern CNII «Jelektropribor» ", 2010, pp.10-13 [in Russian].

[13] Song, Y. , Zhao, W. Multi-agent system rendezvous via refined social system and individual roles. WSEAS Transactions on Systems and Control. Volume 9, Issue 1, 2014, pp. 526-532.

[14] Stefanuk V.L. Local Organization of Intellectual Systems. Fizmatlit, Moscow, 2004. 328 p. [in Russian].

[15] Yogeswaran M. and Ponnambalam S. G. (2010). Swarm Robotics: An Extensive Research Review, Advanced Knowledge Application in Practice, InTech. pp. 259-278.

[16] Yu, H., Jian, J., Wang, Y. Flocking motion of mobile agents with a leader based on distance-dependent adjacency matrix. Lecture Notes in Computer Science. 1st International Conference on Intelligent Robotics and Applications, ICIRA 2008; Wuhan; China; 15 October 2008 through 17 October 2008; Code 74343. Volume 5314 LNAI, Issue PART 1, 2008, pp. 1165-1174.

[17] Zhiguo Shi, Jun Tu, Qiao Zhang, Lei Liu, Junming Wei. A Survey of Swarm Robotics System. Proc. of the Third Intern. Conf. on Advances in Swarm Intelligence, Shenzhen, China, 2012. V.1, pp. 564-572.